Note	CODE1	APPEND_CHAR	CODE1 + APPEND_CHAR	new code added to Dictionary	output
Begin compression	65 ("A")	66 ("B")	"AB"		
"AB" is not found in the dictionary, output code1					65 ("A")
Add it to the dictionary				256 ("AB")	
	66 ("B")	65 ("A")	"BA"		
"BA" is not found in the dictionary, output code1					66 ("B")
Add it to the dictionary				257 ("BA")	
	"A"	"B"	"AB"		
"AB" is found in the dictionary, change code1	"AB"	"A"	"ABA"		
"ABA" not in the dictionary, output code1					256 ("AB")
Add it to dictionary				258 ("ABA")	
	"A"	"B"	"AB"		
"AB" is in the dictionary, change code1	"AB"	"A"	"ABA"		
"ABA" is in the dictionary, change code1	"ABA"	"B"	"ABAB"		
"ABAB" not in the dictionary, output					258 ("ABA")
Add it to dictionary				259 ("ABAB")	
	"B"	"A"	"BA"		
"BA" found in the dictionary	"BA"	"B"	"BAB"		
"BAB" not in the dictionary					257 ("BA")
Add it to the dictionary				260 ("BAB")	
	"B"	"A"	"BA"		
"BA" found	"BA"	"B"	"BAB"		
"BAB" found	"BAB"	"A"	"BABA"		
"BABA" not in the dictionary					260("BAB")
Add it to the dictionary				261("BABA")	
	"A"	"B"	"AB"		
"AB" found	"AB"	"A"	"ABA"		
"ABA" found	"ABA"	"B"	"ABAB"		
"ABAB" found	"ABAB"				
End of file reached. Output CODE1					259("ABAB")

FIGURE 1

Note	CODE1	CODE2	STRING	new code added to Dictionary	Output
Read the first input	65 ("A")				"A"
Read the next code		66 ("B")	"B"		
Output STRING and add new entry to the dictionary				256 ("AB")	"B"
Read the next code	66	256	"AB"		
Output STRING and add new entry to the dictionary				257 ("BA")	"AB"
Read the next code	256	258			
Code 258 is not in the dictionary			"ABA"		
Output STRING and add new entry to the dictionary				258 ("ABA")	"ABA"
Read the next code	258	257	"BA"		
Output STRING and add new entry to the dictionary				259 ("ABAB")	"BA"
Read the next code	257	260			
Code 260 not found in the dictionary			"BAB"		
Output STRING and add new entry to the dictionary				260 ("BAB")	"BAB"
Read the next code	260	259			
Code 259 is in the dictionary			"ABAB"		
Output STRING and add new entry to the dictionary				261("BABA")	"ABAB"
End of input reached.					

FIGURE 2

Note	CODE1	CODE2	CODE1 + CODE2	new code added	output
Begin compression	65 ("A")				
"A" is the best match, output CODE1	65 ("A")				65 ("A")
Find best-matched CODE2	65 ("A")	66("B")	"AB"		
Add it to the dictionary				256("AB")	
Find best-matched CODE1	66 ("B")				
output CODE1	66 ("B")				66 ("B")
Find best-matched CODE2	66 ("B")	256 ("AB")	"BAB"		
Add it to the dictionary				257("BAB")	
Find best-matched CODE1	256 ("AB")				
output CODE1	256 ("AB")				256 ("AB")
Find best-matched CODE2	256 ("AB")	256 ("AB")	"ABAB"		
Add it to the dictionary				258("ABAB")	
Find best-matched CODE1	258("ABAB")				
output CODE1	258("ABAB")				258("ABAB")
Find best-matched CODE2	258("ABAB")	258("AB AB")	"ABABABA B"		
Add it to the dictionary				259("ABABA BAB")	
Find best-matched CODE1	259("ABABA BAB")				
output CODE1	259("ABABA BAB")				259("ABABA BAB")
No more input, end of compression					

FIGURE 3

Note	COD E1	COD E2	new code added	Output
Read the first input	65 ("A")			
Yes, there is more input	65 ("A")			
Output string defined by CODE1	65 ("A")			66("A")
Read the next code, CODE2	65 ("A")	66 ("B")		
CODE2 is in the dictionary, add CODE1::CODE2	65 ("A")	66 ("B")	256("AB")	
CODE1 <- CODE2	66 ("B")			
Yes, there is more input	66 ("B")			
Output string defined by CODE1	66 ("B")			66 ("B")
Read the next code, CODE2	66 ("B")	256		
CODE2 is in the dictionary, add CODE1::CODE2	66 ("B")	256	257("BAB")	
CODE1 <- CODE2	256			
Yes, there is more input	256			
Output string defined by CODE1	256			256 ("AB")
Read the next code, CODE2	256	258		
CODE2 is not in the dictionary, add CODE1::CODE1	256	258	258("ABAB")	
CODE1 <- CODE2	258			
Yes, there is more input	258			
Output string defined by CODE1	258			258("ABAB")
Read the next code, CODE2	258	259		
CODE2 not in the dictionary, add CODE1::CODE1	258	259	259("ABABABAB")	
CODE1 <- CODE2	259			
No more input	259			
Output string defined by CODE1	259			259("ABABABAB")
End of decompression				

FIGURE 4

Compression Engine		Decompression Engine	
Output Transmitted	Vocabulary Added	Input Received	Vocabulary Added
65 ("A")			
	256("AB")	65 ("A")	
66 ("B")			
	257("BAB")	66 ("B")	
			256("AB")
256 ("AB")			
	258("ABAB")	256 ("AB")	
			257("BAB")
258("ABAB")			
	259("ABABABAB")	258("ABAB")	
			258("ABAB")
259("ABABABAB")			
		259("ABABABAB")	
			259("ABABABAB")

FIGURE 5

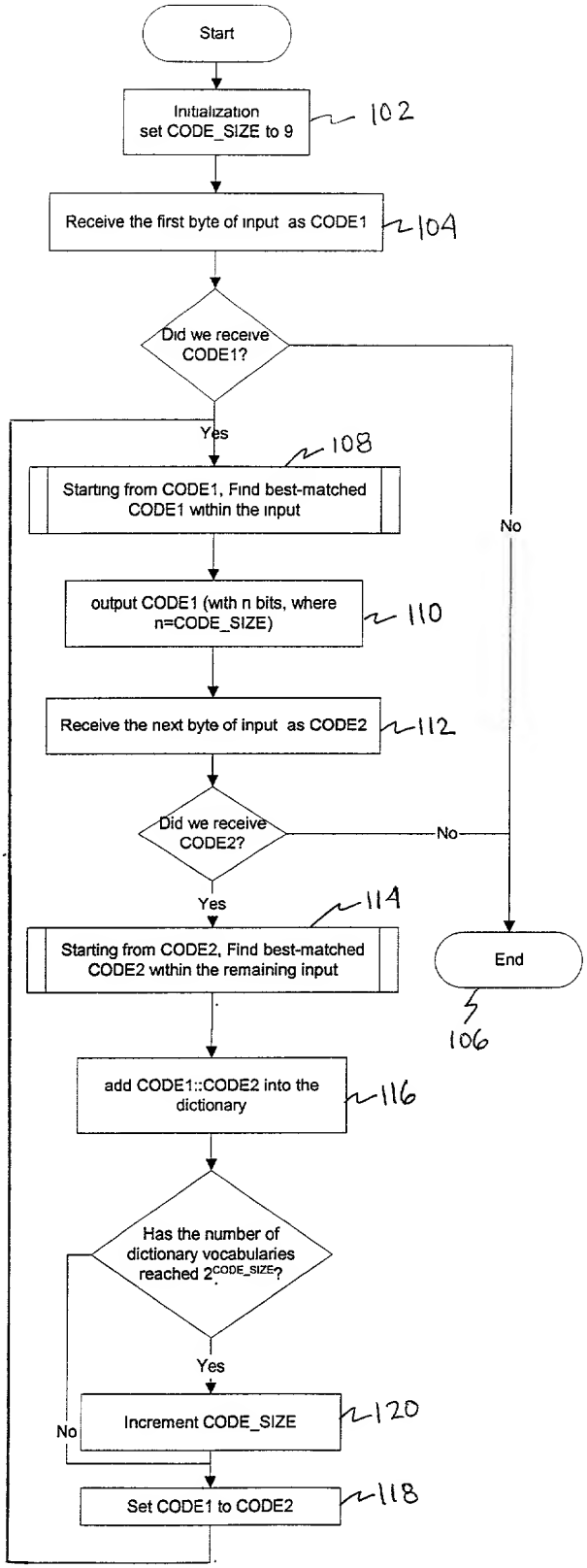

Start
Initialization
set CODE_SIZE to 9
102
Receive the first byte of input as CODE1
104
Did we receive CODE1?
Yes
No
108
Starting from CODE1, Find best-matched CODE1 within the input
output CODE1 (with n bits, where n=CODE_SIZE)
110
Receive the next byte of input as CODE2
112
Did we receive CODE2?
No
Yes
114
Starting from CODE2, Find best-matched CODE2 within the remaining input
End
106
add CODE1::CODE2 into the dictionary
116
Has the number of dictionary vocabularies reached 2^CODE_SIZE?
Yes
No
Increment CODE_SIZE
120
Set CODE1 to CODE2
118

FIGURE 6A

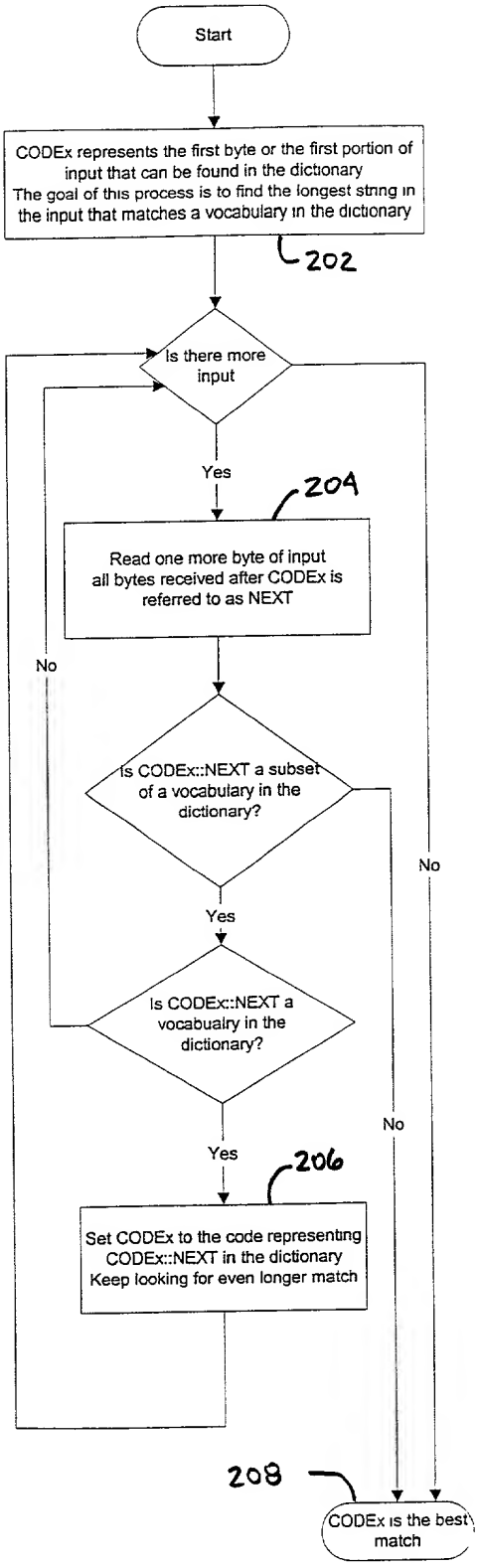
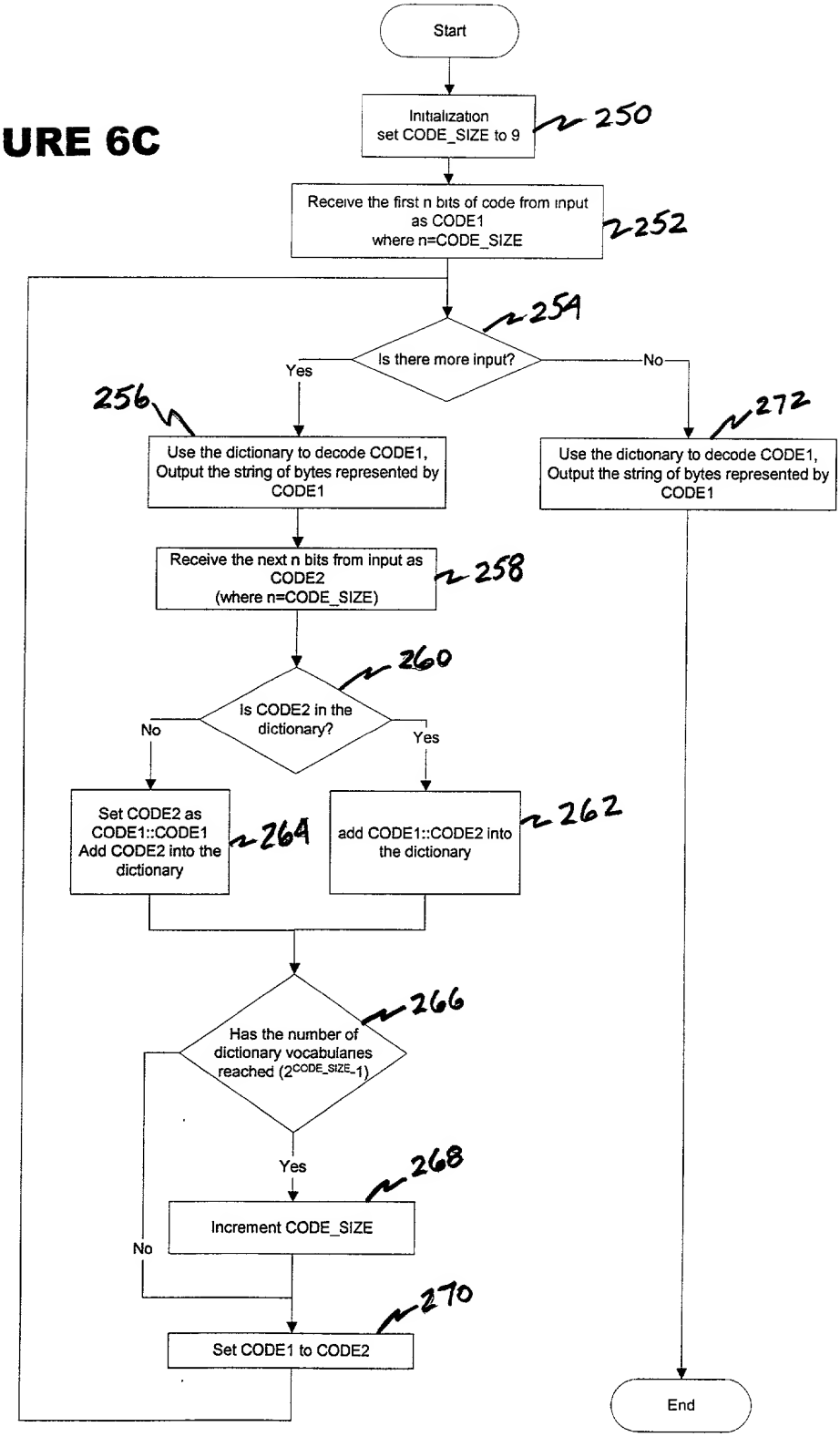

Start
CODEx represents the first byte or the first portion of input that can be found in the dictionary
The goal of this process is to find the longest string in the input that matches a vocabulary in the dictionary
202
Is there more input
Yes
204
Read one more byte of input all bytes received after CODEx is referred to as NEXT
No
Is CODEx::NEXT a subset of a vocabulary in the dictionary?
No
Yes
Is CODEx::NEXT a vocabulary in the dictionary?
No
Yes
206
Set CODEx to the code representing CODEx::NEXT in the dictionary
Keep looking for even longer match
208
CODEx is the best match

FIGURE 6B

FIGURE 6C


Start
Initialization
set CODE_SIZE to 9
250
Receive the first n bits of code from input
as CODE1
where n=CODE_SIZE
252
254
Is there more input?
Yes
No
256
Use the dictionary to decode CODE1,
Output the string of bytes represented by
CODE1
272
Use the dictionary to decode CODE1,
Output the string of bytes represented by
CODE1
Receive the next n bits from input as
CODE2
(where n=CODE_SIZE)
258
260
Is CODE2 in the
dictionary?
No
Yes
Set CODE2 as
CODE1::CODE1
Add CODE2 into the
dictionary
264
add CODE1::CODE2 into
the dictionary
262
266
Has the number of
dictionary vocabularies
reached (2^CODE_SIZE-1)
Yes
268
Increment CODE_SIZE
No
270
Set CODE1 to CODE2
End

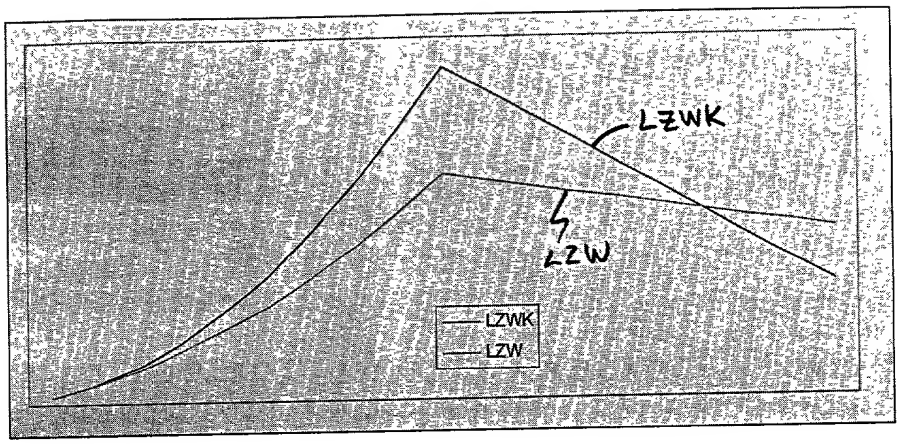


Figure 7

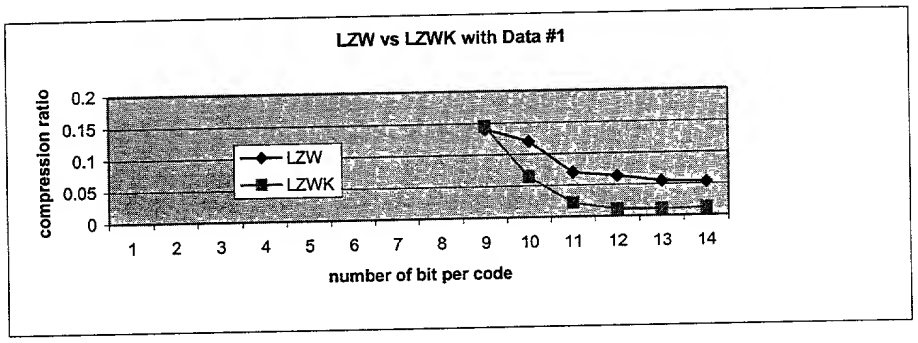


Figure 8

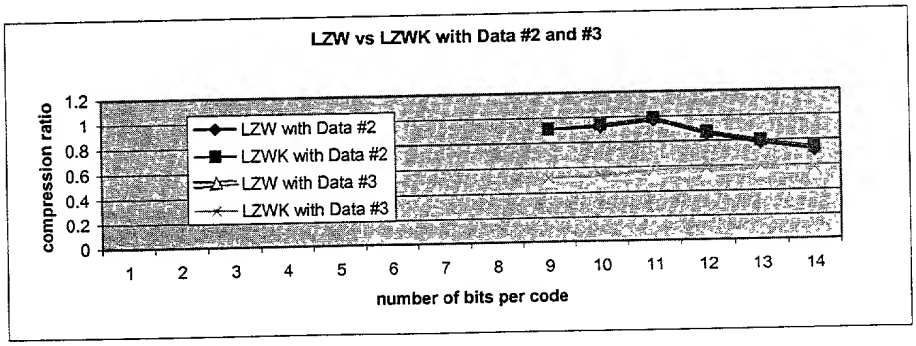


Figure 9